

# **Programmation avancée**

## **TP07 : CRUD, Formulaires et FlashBag**

# Table des matières

|                                   |    |
|-----------------------------------|----|
| Introduction.....                 | 3  |
| CRUD de l'Entité Article.....     | 3  |
| Variables globales dans Twig..... | 5  |
| FlashBag.....                     | 6  |
| Amélioration de l'IHM.....        | 7  |
| Conclusion.....                   | 11 |

# Introduction

Dans ce TP nous allons créer un CRUD, c'est-à-dire une interface pour modifier/ajouter/supprimer les articles.

## CRUD de l'Entité Article

Pour faire fonctionner la création de nouveaux articles, voici le code que j'ai écrit :

```
public function buildForm(FormBuilderInterface $builder)
{
    $builder
        ->add('title', TextType::class, [
            'label' => 'Titre',
            'required' => true,
        ])
        ->add('content', TextType::class, [
            'label' => 'Contenu',
            'required' => true,
        ])
        ->add('imageUrl', UrlType::class, [
            'label' => 'URL image',
            'required' => true,
        ])
        ->add('createdAt', DateType::class, [
            'input' => 'datetime_immutable',
            'label' => 'Date de création',
            'required' => true,
        ])
        ->add('updatedAt', DateType::class, [
            'input' => 'datetime_immutable',
            'label' => 'Date de mise à jour',
            'required' => true,
        ])
        ->add('slug', TextType::class, [
            'required' => true,
        ])
        ->add('author', EntityType::class, [
            'class' => User::class,
            'choice_label' => 'name',
            'multiple' => false,
            'expanded' => false,
            'required' => true,
            'label' => 'Auteur',
        ])
    }
```

```

->add('categories', EntityType::class, [
    'class' => Category::class,
    'choice_label' => 'name',
    'multiple' => true,
    'expanded' => true,
    'required' => true,
    'label' => 'Categories',
])
;
}

```

Et voici le résultat :

## Create new Article

Titre

Contenu

URL image

Date de création

Jan ▼ 1 ▼ 2018 ▼

Date de mise à jour

Jan ▼ 1 ▼ 2018 ▼

Slug

Auteur Charles Simon ▼

Categories

☐ Atque aut sequi. ☐ Porro qui nemo. ☐ Tenet voluptate architecto est.

L'ajout/modification et suppression d'un article marche bien.

## Variables globales dans Twig

Après avoir ajouté les variables globales j'ai décommenté la génération des liens de catégories et utilisé la variable globale pour générer la liste des catégories sur toute la page sans avoir à faire de modification :

```
{% for category in categoryRepository.findAll() %}
    <li class="nav-item"><a href="{{path('app_articles_by_category', {
        {{category.name}}}}</a>
        ({{category.articles | length}} articles)
    </li>
{% endfor %}
```

Résultat :



The screenshot shows a web browser at the URL `193.54.227.208/~syfy20/symfony/public/articles/new`. The page has a dark blue header with the text "Blog GUIBERT" and navigation links "ABOUT", "CATEGORIES", and "CONTACT". The main content area is split into two columns. The left column, titled "Create new Article", contains a form with the following fields: "Titre" (text input), "Contenu" (text input), "URL image" (text input), "Date de création" (date picker set to Jan 1, 2018), and a "Create" button. The right column displays a list of categories with their respective article counts:

- ATQUE AUT SEQUI.(20 ARTICLES)
- PORRO QUI NEMO.(18 ARTICLES)
- TENETUR MAIORES IN... (20 ARTI
- TENETUR IPSA SUNT.(16 ARTICLE
- VOLUPTATE ARCHITEC... (26 ARTI

# FlashBag

Il faut ajouter le Flash dans les méthodes d'ajout/modification/modification d'article :

```
#[Route('/new', name: 'app_article_new', methods: ['GET', 'POST'])]  
public function new(Request $request, ArticleRepository $articleRepos  
{  
    $article = new Article();  
    $form = $this->createForm(ArticleType::class, $article);  
    $form->handleRequest($request);  
  
    if ($form->isSubmitted() && $form->isValid()) {  
        $this->addFlash('notice', 'Article ajouté!');  
  
        $articleRepository->save($article, true);  
    }  
}
```

```
#[Route("/{id}/edit", name: 'app_article_edit', methods: ['GET', 'P  
public function edit(Request $request, Article $article, ArticleRep  
{  
    $form = $this->createForm(ArticleType::class, $article);  
    $form->handleRequest($request);  
  
    if ($form->isSubmitted() && $form->isValid()) {  
        $this->addFlash('notice', 'Article modifié!');  
  
        $articleRepository->save($article, true);  
    }  
}
```

```
#[Route("/{id}", name: 'app_article_delete', methods: ['POST'])]  
public function delete(Request $request, Article $article, Article  
{  
    $this->addFlash('notice', 'Article supprimé!');  
  
    if ($this->isCsrfTokenValid('delete' . $article->getId(), $reque  
        $articleRepository->remove($article, true);  
    }  
}
```

Ajout:



Modification :



Suppression :




# Amélioration de l'IHM





Icônes pour les actions :

|              |     |     |              |                     |                     |       |   |
|--------------|-----|-----|--------------|---------------------|---------------------|-------|---|
| 305          | sdf | sdf | http://sdfsd | 2018-01-01 00:00:00 | 2018-01-01 00:00:00 | sdfsd |   |
| ⊕ Create new |     |     |              |                     |                     |       |   |







Lien d'ajout en haut :

| Article index |                             |   |   |                     |           |                            |   |
|---------------|-----------------------------|---|---|---------------------|-----------|----------------------------|---|
| ⊕ Create new  |                             |   |   |                     |           |                            |   |
| Id            | Title                       | Content   | ImageUrl                                | CreatedAt           | UpdatedAt | Slug                       | ac  |
| 201           | Earum qui officia quo quia. | Officiis eos inventore nihil deleniti rem temporibus. | https://picsum.photos/360/360?image=300 | 2016-12-26 11:46:51 |           | earum-qui-officia-quo-quia |  |



Contours au tableau :

| Article index |                             |   |   |                     |           |                            |   |
|---------------|-----------------------------|---|---|---------------------|-----------|----------------------------|---|
| ⊕ Create new  |                             |   |   |                     |           |                            |   |
| Id            | Title                       | Content   | ImageUrl                                | CreatedAt           | UpdatedAt | Slug                       | actions   |
| 201           | Earum qui officia quo quia. | Officiis eos inventore nihil deleniti rem temporibus. | https://picsum.photos/360/360?image=300 | 2016-12-26 11:46:51 |           | earum-qui-officia-quo-quia |   |
| 202           | Dolores veniam libero.      | Repellendus distinctio                                | https://picsum.photos/360/360?image=301 | 2011-04-04          |           | dolores-veniam-libero      |   |

Images à la place des URL :

| ⊕ Create new |                             |   |   |                     |                     |                            |   |
|--------------|-----------------------------|---|---|---------------------|---------------------|----------------------------|---|
| Id           | Title                       | Content   | ImageUrl  | CreatedAt           | UpdatedAt           | Slug                       | actions   |
| 201          | Earum qui officia quo quia. | Officiis eos inventore nihil deleniti rem temporibus. |  | 2018-12-26 00:00:00 | 2018-01-01 00:00:00 | earum-qui-officia-quo-quia |   |
| 202          | Dolores veniam libero.      | Repellendus distinctio                                |  | 2011-04-04          |                     | dolores-veniam-libero      |   |

Formatage des dates en français :

| Article index |                               |   |   |                     |                     |                |
|---------------|-------------------------------|---|---|---------------------|---------------------|----------------|
| ⊕ Create new  |                               |   |   |                     |                     |                |
| Id            | Title                         | Content   | ImageUrl  | CreatedAt           | UpdatedAt           |                |
| 201           | Earum qui officia quo quia.   | Officiis eos inventore nihil deleniti rem temporibus. |  | 26-12-2018 00:00:00 | 01-01-2018 00:00:00 | ea<br>of       |
| 202           | Dolores veniam libero magnam. | Repellendus distinctio perferendis perspiciatis.      |  | 04-04-2011 06:13:47 |                     | dc<br>ve<br>m: |

Catégories d'un article dans la vue « show » :

| Article           |   |
|-------------------|---|
| <b>Id</b>         | 299   |
| <b>Title</b>      | Eaque quis et.                              |
| <b>Content</b>    | Omnis tempore sit est eos minus et dolores. |
| <b>ImageUrl</b>   | https://picsum.photos/360/360?image=398     |
| <b>CreatedAt</b>  | 1991-10-07 19:21:18                         |
| <b>UpdatedAt</b>  |   |
| <b>Slug</b>       | eaque-quis-et                               |
| <b>Categories</b> | Atque aut sequi.                            |

Champs dans une table HTML :

| Create new Article   |  |
|----------------------|--|
| Titre                | <input type="text"/>   |
| Contenu              | <input type="text"/>   |
| URL image            | <input type="text"/>   |
| Date de création     | Jan ▼ 1 ▼ 2018 ▼   |
| Date de modification | Jan ▼ 1 ▼ 2018 ▼   |
| Slug                 | <input type="text"/>   |
| Auteur               | Charles Simon ▼  |
| Categories           | <input type="checkbox"/> Atque aut sequi. <input type="checkbox"/> Porro qui nemo. <input type="checkbox"/> Tenetur maiores incidunt. <input type="checkbox"/> Tenetur ipsa sunt. <input type="checkbox"/> Voluptate architecto est. |



Ne pas afficher le slug :

## Edit Article







|                      |   |
|----------------------|---|
| Titre                | <input type="text" value="c'est le titre"/>   |
| Contenu              | <input type="text" value="sfsd"/>   |
| URL image            | <input type="text" value="http://sdfs"/>  |
| Date de création     | Jan 1 2018  |
| Date de modification | Dec 10 2023   |
| Auteur               | Charles Simon   |
| Categories           | <input type="checkbox"/> Atque aut sequi. <input type="checkbox"/> Porro qui nemo. <input type="checkbox"/> ipsa sunt. <input type="checkbox"/> Voluptate architecto est. |

Update

Mise à jour de updatedAt :

```
#[Route('/{id}/edit', name: 'app_article_edit', methods: ['GET', 'POST'], public function edit(Request $request, Article $article, ArticleRepository $articleRepository) {  
    $form = $this->createForm(ArticleType::class, $article);  
    $form->handleRequest($request);  
  
    if ($form->isSubmitted() && $form->isValid()) {  
        $this->addFlash('notice', 'Article modifié !');  
  
        $article->setUpdatedAt(new DateTimeImmutable());  
        $articleRepository->save($article);  
    }  
}
```

Create new

| Id  | Title                         | Content   | ImageUrl  | CreatedAt           | UpdatedAt           | Slug                         | actions   |
|-----|-------------------------------|---|---|---------------------|---------------------|------------------------------|---|
| 201 | Earum qui officia quo quia.   | Officiis eos inventore nihil deleniti rem temporibus. |  | 26-12-2018 00:00:00 | 09-12-2023 17:11:19 | earum-qui-officia-quo-quia   |   |
| 202 | Dolores veniam libero magnam. | Repellendus distinctio perferendis perspiciatis.      |  | 04-04-2018 00:00:00 | 09-12-2023 17:11:56 | dolores-veniam-libero-magnam |   |
|     |                               | Architecto  |   |                     |                     |                              |   |

Mise à jour du slug :

```
#[Route('/new', name: 'app_article_new', methods: ['GET', 'POST'])]
public function new(Request $request, ArticleRepository $articleRepository)
{
    $article = new Article();
    $form = $this->createForm(ArticleType::class, $article);
    $form->handleRequest($request);

    if ($form->isSubmitted() && $form->isValid()) {
        $this->addFlash('notice', 'Article ajouté !');

        $article->setSlug($article->getTitle());
        $articleRepository->save($article, true);
    }
}
```

```
#[Route("/{id}/edit", name: 'app_article_edit', methods: ['GET', 'POST'])]
public function edit(Request $request, Article $article, ArticleRepository $articleRepository)
{
    $form = $this->createForm(ArticleType::class, $article);
    $form->handleRequest($request);

    if ($form->isSubmitted() && $form->isValid()) {
        $this->addFlash('notice', 'Article modifié !');

        $article->setUpdatedAt(new DateTimeImmutable());
        $article->setSlug($article->getTitle());
        $articleRepository->save($article, true);
    }
}
```

## Conclusion

Dans ce TP on a créé un CRUD pour les articles et avons utilisé les variables globales ainsi qu'améliorer l'interface.